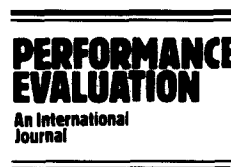




ELSEVIER

Performance Evaluation 19 (1994) 279–302



# Fixed point iteration on pointsymmetric interference graphs

J. Peter Veltkamp <sup>a,\*</sup>, Ruud van Damme <sup>b</sup>

<sup>a</sup> *Dept. of Mathematics and Computing Science, Eindhoven University of Technology, P.O. Box 513,  
5600 MB Eindhoven, Netherlands*

<sup>b</sup> *Dept. of Applied Mathematics, University of Twente, P.O. Box 217, 7500 AE Enschede, Netherlands*

(Received 15 December 1990; revised 11 June 1993)

---

## Abstract

Interference graphs are used for performance analysis of multiprocessor interconnection networks. In order to model blocked transmissions, nodes can have three states: idle, active or blocked. The resulting steady state probability distribution has a non-product form. Macro states are introduced to calculate performance measures, and the corresponding macro state probability distribution is approximated by a special type of fixed point iteration: the macro approximation, which is very efficient for pointsymmetric interference graphs.

*Key words:* Birth-death process; Performance analysis; Fixed point iteration; Interference graph; Pointsymmetry

---

## 1. Introduction

Interference graphs were introduced by Yemini and Pinsky [7] and [4] to analyse the performance of multiprocessor interconnection networks. For applications of interference graphs the reader is referred to these papers. Here we focus on the model itself, which is extended in such a way that blocked transitions are modelled explicitly. The emphasis is on the mathematical approximation technique and not on the particular deblocking strategy that we choose to compare theoretical and simulation results. First we describe the original model and then we give our extension. A processor is called active if it is sending a message through the network and otherwise idle. A communication path through the network between two processors is called active if it is transmitting a message and otherwise idle. Depending on whether one is interested in the average number of active processors or the average number of active communication paths, either the processors or the communication paths are mapped onto the nodes of an interference graph. An edge between two nodes of an interference graph means that these nodes cannot be active simultaneously, i.e., only one of them can be active at a time.

---

\* Corresponding author. Email: [wsinpv@win.tue.nl](mailto:wsinpv@win.tue.nl).

This happens for instance when two processors are connected to the same communication channel and only one message at a time can be sent through the channel or when two communication paths in an interconnection network cross each other. The average number of active nodes is calculated assuming that nodes can make random transitions from the idle to the active state and vice versa. Suppose an idle node becomes active according to an exponential inter arrival law with rate  $\lambda$  and that an active period lasts an exponentially distributed time with rate  $\mu$ . Define  $\Lambda$  as the set of nodes in the interference graph. We say two nodes are neighbours of each other if they are connected by an edge. Edges can only exist between different nodes and have no direction. At most one edge can exist between a pair of different nodes. If  $X \subseteq \Lambda$  then we define the boundary  $\partial X$  of  $X$  as the set of nodes in  $\Lambda \setminus X$  that have at least one neighbour in  $X$ . If we write the neighbour relation between two nodes  $x$  and  $y$  as  $x \sim y$ , we can define  $\partial X$  formally as:

$$\partial X \equiv \{x \in \Lambda \setminus X \mid \exists_{y \in X} x \sim y\}$$

The closure  $\bar{X}$  of  $X$  is defined as  $\bar{X} = X \cup \partial X$  and the complement of this closure as  $\bar{X}^c \equiv \Lambda \setminus \bar{X}$ . Let  $A$  be a set of nodes that can be active simultaneously, i.e.

$$A \subseteq \Lambda \wedge (\forall x, y: x, y \in A: \neg(x \sim y))$$

We can now write down the equation for the steady state probability  $p(A)$  that the set of nodes  $A$  is active and the set  $\Lambda \setminus A$  is idle:

$$p(A) [\lambda |\bar{A}^c| + \mu |A|] = \sum_{x \in \bar{A}^c} \mu p(A \cup \{x\}) + \sum_{x \in A} \lambda p(A \setminus \{x\}),$$

where we have denoted the number of elements in a finite set  $X$  by  $|X|$ . The first term at the left-hand side of this equation denotes the rate out of state  $A$  to states that contain one more active node, whereas the second term at the left-hand side denotes the rate out of  $A$  to states with one active node less. The first term at the right-hand side denotes the rate into state  $A$  from states with one more active node, and the second term denotes the rate into  $A$  from states with one active node less. The state space  $S$  consists by definition of all the subsets of  $\Lambda$  that contain no neighbouring nodes. The solution of the equation above can now be given as:

$$p(A) = \begin{cases} \rho^{|A|} / Z & \text{if } A \in S \\ 0 & \text{otherwise} \end{cases}$$

where  $\rho \equiv \lambda / \mu$  and  $Z \equiv \sum_{A \in S} \rho^{|A|}$ . The function  $Z(\rho)$  is called the *partition function* of the interference system. The solution for  $p(A)$  can easily be checked by direct substitution into the equilibrium equation. We say that  $p(A)$  exhibits a product form as it seems to factorise to the ratios  $\rho = \lambda / \mu$  denoting the time fraction that a node would be active if it were in isolation without interferences, for all nodes. We call this model for an interference system the Yemini-model after Y. Yemini who introduced it in 1982 [7]. From this product form solution we can deduce directly that there must be detailed balance:

$$\lambda p(A) = \mu p(A \cup \{x\}),$$

for all  $A \in S$  and  $x \in \bar{A}^c$ . In general one can always deduce the equilibrium equation from the

detailed balance equation, but the reverse is not true. Of prime interest in the Yemini-model is the calculation of the average concurrency  $E$  in the interference system, defined as:

$$E \equiv \sum_{A \in S} |A| p(A).$$

This can also be written as:

$$E = \sum_{A \in S} |A| \rho^{|A|} / Z = \rho \frac{\partial \log z(\rho)}{\partial \rho}.$$

From  $E$  some other performance measures can be deduced, namely the throughput  $T \equiv \mu E$  and the probability  $B \equiv T/N\lambda$  that an arriving activity is not blocked, where  $N$  stands for the number of components in the interference system. The throughput is in this case equal to the average number of concurrent activities processed by the system per unit time. Of course  $T$  and  $B$  can also be derived directly from  $Z(\rho)$ :

$$T = \lambda \frac{\partial \log Z(\rho)}{\partial \rho}$$

and

$$B = \frac{1}{N} \frac{\partial \log Z(\rho)}{\partial \rho},$$

so the problem of calculating these performance measures reduces to the calculation of the partition function  $Z(\rho)$ . Yemini remarks that this is reminiscent of statistical mechanics, where all the thermodynamic observables can be derived from a partition function. The average concurrency  $E$  can be interpreted as the average energy of the interference system, because if we define an inverse temperature  $\beta = -\log \rho$ , the partition function can be rewritten as:

$$Z(\beta) = \sum_{A \in S} e^{-\beta |A|},$$

so that  $E$  becomes equal to

$$E = - \frac{\partial \log Z(\beta)}{\partial \beta}.$$

The main deficiency of the Yemini-model is that it cannot account for blocked transmissions, i.e., if an idle node in the Yemini-model receives an activation request which cannot be satisfied, the node stays idle, whereas in reality such a node enters the blocked (or back-off) state until it gets permission to become active. In order to model blocking, we extend the Yemini-model with a blocked state so that each node is always in one of the three states idle ( $I$ ), active ( $A$ ) or blocked ( $B$ ). We call the resulting model the *LAB*-model, which will be discussed in the next section. Instead of the equilibrium probability  $p(A)$  we now get  $p(A; B)$  for the probability that the nodes in set  $A$  are active and in set  $B$  are blocked, whereas the rest of them ( $\Lambda \setminus (A \cup B)$ ) is idle. The problem in calculating  $p(A; B)$  is its non-product form due to the irreversible nature of the *LAB*-model. Fortunately we do not need to know  $p(A; B)$  in order to calculate the relevant performance measures, as will be demonstrated in Section 2.

These performance measures can be expressed solely in terms of so-called macro probabilities of macro states in which the states of only a few nodes are specified instead of the states of all the nodes in the interference graph. In Sections 3 and 4 we describe an approximation for these macro probabilities which is a generalisation of the approximation technique given by Brandwajn [2]. The generalisation consists of two aspects. First, to calculate the interaction of a given node with its environment, we do not only take into account the nearest neighbours of that node, but also the next-nearest neighbours or more. Second, we also consider interference graphs with more than two neighbours per node, whereas Brandwajn considered only two. Another important difference with Brandwajn is our restriction to pointsymmetric interference graphs. This is not an essential restriction of our approximation technique, but it reduces the computational complexity dramatically. In Section 5 we present our results for three pointsymmetric interference graphs, i.e., the ring, triangular dodecahedron and torus. Finally, in Section 6 we present our conclusions.

## 2. Extension of the Yemini-model

### 2.1. Equilibrium equations

In this section we will extend the Yemini-model to three states per node: idle, active and blocked. When an idle node received an unsatisfiable activation request in the Yemini-model, the request was simply rejected. Now we assume that in such cases the idle node makes a transition from the idle to the blocked state. As soon as the blocked node has no active neighbours anymore, it becomes active. If an active node keeps more than one neighbour blocked, all blocked neighbours become active simultaneously as soon as the active node becomes idle, unless the blocked neighbours have other active neighbours or are neighbours of each other. In the former case the blocked node stays blocked until it has no active neighbours anymore, in the latter a choice must be made which nodes may become active and which ones have to stay blocked. In this model we will only assume that the deblocking strategy is stochastic, so that the model remains a spatial birth-death process. The state space now consists of ordered pairs  $(A; B)$ , where  $A$  denotes the set of active nodes in  $\Lambda$  and  $B$  the set of blocked nodes in  $\Lambda$ , whereas all other nodes in  $\Lambda$  are idle. If each idle node becomes active (or blocked) with rate  $\lambda$  and if each active node becomes idle with rate  $\mu$ , then the equilibrium probability distribution  $p(A; B)$  must satisfy the following set of equations:

$$\begin{aligned}
 & p(A; B) \{ \lambda (|\bar{A}^c| + |\partial A \setminus B|) + \mu |A| \} \\
 &= \lambda \left\{ \sum_{a \in A} p(A \setminus \{a\}; B) + \sum_{b \in B} p(A; B \setminus \{b\}) \right\} \\
 &+ \mu \left\{ \sum_{a \in \bar{A}^c} p(A \cup \{a\}; B) + \sum_{\substack{a \in \partial A \setminus B \\ B \subseteq \partial(\{a\} \cup A \setminus (A \cap \partial(a)))}} \sigma_{(A \setminus (A \cap \partial(a)) \cup \{a\}; B \cup (A \cap \partial(a)))} (a | A \cap \partial(a)) \right. \\
 &\quad \left. \times p(\{a\} \cup A \setminus (A \cap \partial(a)); B \cup (A \cap \partial(a))) \right\}.
 \end{aligned}$$

The first term on the left-hand side denotes the departure rate from  $(A; B)$  due to activation of nodes in the complement of the closure of  $A$  ( $\bar{A}^c = \Lambda \setminus (A \cup \partial A)$ ); the second term on the left-hand side is the departure rate from  $(A; B)$  due to blocking of idle nodes in  $\partial A$  and the third term on the left-hand side is the departure rate from  $(A; B)$  due to de-activation of nodes in  $A$ ; the terms on the right-hand side are all arrival rates in  $(A; B)$ : the first term is the arrival rate due to activation of idle nodes in  $A$ ; the second term is the arrival rate due to blocking of idle nodes in  $B$ ; the third term is the arrival rate due to de-activation of nodes in  $\bar{A}^c$  and the fourth term is the arrival rate due to the simultaneous deblocking of the nodes in  $A \cap \partial\{a\}$  when node  $a$  becomes idle;  $a$  must be an element of  $\partial A \setminus B$  and the active nodes in the departure state must be capable of keeping the nodes in  $B$  blocked, while  $a$  is still active, i.e.,  $B \subseteq \partial(\{a\} \cup A \setminus (A \cap \partial\{a\}))$ . And  $\sigma_{(A \setminus (A \cap \partial\{a\}) \cup \{a\}; B \cup (A \cap \partial\{a\}))}(a | A \cap \partial\{a\})$  denotes the probability that when node  $a$  in the active set  $A \setminus (A \cap \partial\{a\}) \cup \{a\}$  becomes idle, the set  $A \cap \partial\{a\}$  in the blocked set  $B \cup (A \cap \partial\{a\})$  becomes active. The probabilities  $\sigma$  determine the stochastic deblocking strategy.

It is obvious that this extended Yemini-model must be irreversible: idle nodes can become blocked and blocked nodes can become active, but active nodes can never become blocked and blocked nodes can never become idle.

## 2.2. Performance measures

In order to calculate the relevant performance measures in the IAB-model, we introduce the concept of a *macro state*. In a macro state we do not specify the states of all the nodes, but only a subset of them. We write macro states as  $[A; B; I]$ , where  $A$  denotes the set of active nodes,  $B$  the set of blocked nodes and  $I$  the set of idle nodes. Since not all the nodes need to be specified, we have  $A \cup B \cup I \subseteq \Lambda$ . In contrast, we call a state  $(A; B)$  where all the nodes are specified a *micro state*. Note that a micro state is a special case of a macro state. The micro state probability distribution  $p(A; B)$  induces a probability distribution  $\pi[A; B; I]$  on macro state space. In particular we have  $p(A; B) = \pi[A; B; \Lambda \setminus (A \cup B)]$ , otherwise the probability of a macro state is given by the sum of the probabilities of all micro states that are compatible with that macro state. The average number of active nodes  $\langle |A| \rangle$  can now be calculated as follows:

$$\begin{aligned} \langle |A| \rangle &= \sum_{(A;B) \in S} |A| p(A; B) = \sum_{(A;B) \in S} \sum_{i \in \Lambda} f_A(i) p(A; B) \\ &= \sum_{i \in \Lambda} \sum_{(A;B) \in S} f_A(i) p(A; B) = \sum_{i \in \Lambda} \langle f_A(i) \rangle \\ &= \sum_{i \in \Lambda} \pi[\{i\}; \emptyset; \emptyset], \end{aligned}$$

where  $S$  denotes the state space of all micro states and  $f_A(i)$  is an indicator function defined by:

$$f_A(i) = \begin{cases} 1 & \text{if } i \in A \\ 0 & \text{if } i \notin A \end{cases}$$

The average number of idle nodes  $\langle |I| \rangle$  and blocked nodes  $\langle |B| \rangle$  can similarly be expressed in macro probabilities:

$$\langle |B| \rangle = \sum_{i \in \Lambda} \pi[\emptyset; \{i\}; \emptyset]$$

$$\langle |I| \rangle = \sum_{i \in \Lambda} \pi[\emptyset; \emptyset; \{i\}]$$

but there is more. From Little's theorem follows that

$$\langle |A| \rangle = \rho \langle |I| \rangle.$$

Further we have  $\langle |A| \rangle + \langle |B| \rangle + \langle |I| \rangle = N$  (the total number of nodes) so that

$$\langle |B| \rangle = N - (1 + \rho) \langle |I| \rangle.$$

Knowledge of  $\langle |A| \rangle$  is thus sufficient to calculate the other two.

We are also interested in the average time  $\tau_B$  that a blocked node spends in the blocked state before it becomes active. For a Markov process with transition matrix  $Q_{i,j}$  the average lifetime of state  $i$  is given by

$$\tau = \frac{1}{-Q_{i,i}} \equiv \frac{1}{\sum_{j \neq i} Q_{i,j}}.$$

The time-evolution of the probability  $\xi_i(t)$  that node  $i$  is blocked at time  $t$ , given that it was blocked at time 0, is not governed by a Markov process however. This means that  $\tau_B$  cannot be expressed in time-independent macro probabilities. In principle this problem can be overcome in our approximation by solving a set of non-linear first order coupled differential equations. It is much simpler, however, to make an additional approximation by assuming that 1-node macro states evolve according to Markov processes, so that  $\xi_i(t)$  becomes an exponential distribution. In that case  $\tau_B$  can be expressed in the macro transition rate  $\bar{Q}_{BA}$  for a blocked node to become active:

$$\tau_B = \bar{Q}_{BA}^{-1},$$

and this can again be expressed in time-independent macro probabilities, as will be shown in the next section.

### 3. Macro equations

We now discuss a direct approximation of the macro probabilities: the macro approximation. In principle this approximation can handle infinitely large systems, since the macro probabilities are approximated directly, thereby circumventing the problem of exponentially large summations over states of unspecified nodes. But there is one important restriction on the applicability of this approximation: *the interference graph must be pointsymmetric* (this demand can be relaxed to “almost pointsymmetric”). Before we explain what we mean by this, we first

remark that the macro approximation implicitly assumes that the relevant macro probabilities (with respect to performance measures) are independent of the size of the interference graph. This assumption implies amongst other things that the throughput per node must be independent of the number of nodes. In practice the throughput per node seems to have a limit if the number of nodes approaches infinity, so that the assumption seems valid for interference graphs that are sufficiently large.

### 3.1. Pointsymmetry

How do we define a pointsymmetric graph? Roughly, we call a graph pointsymmetric if it “looks the same” from every node. To make this definition more precise, we first label the nodes  $1, \dots, N$ . With this labelling we can define a connectivity matrix  $C_{i,j}$  by  $C_{i,j} = 1$  if node  $i$  and  $j$  are connected by an edge and  $C_{i,j} = 0$  if this is not the case (for  $i = j$  the definition is arbitrary; we take  $C_{i,i} = 1$ ). Now suppose we make a new labelling, where any of the old nodes  $2, \dots, N$  can become the new number 1. With every new labelling corresponds a new connectivity matrix  $C'_{i,j}$ . For every new number 1 there are still many labellings possible for the remaining  $N - 1$  nodes. If there exists for every new number 1 at least one labelling of the remaining  $N - 1$  nodes, such that  $C'_{i,j} = C_{i,j}$  (for all  $i, j$ ), then we call the graph pointsymmetric. In mathematical literature there exist also other names for pointsymmetry. The concept is there defined with automorphisms and other terminology from group theory (see for instance [1]), but here we have refrained from such a formal approach to support the readers who are unfamiliar with such terminology.

### 3.2. $d$ -simplices

Before we can introduce the macro equations, we must first define what we mean by a simplex of diameter  $d$ , or a  $d$ -simplex. A  $d$ -simplex is defined as a subset of the set of nodes of an interference graph, such that the distance between every pair of nodes in the subset is smaller than or equal to  $d$ . The distance between two nodes in a connected graph is defined as the number of edges on the shortest path that connects both nodes.

### 3.3. Notation

Next we must introduce some new notation in order to write down the macro equations. We used to write  $(A; B)$  for a micro state and  $[A; B; I]$  for a macro state. We will denote the state of a  $d$ -simplex  $s$  by  $[A^s; B^s; I^s]$  with:

$$A^s \equiv A \cap s, B^s \equiv B \cap s, I^s \equiv I \cap s,$$

if the micro state of the whole interference graph is given by  $(A; B)$  with  $I \equiv \Lambda \setminus (A \cup B)$ . Furthermore we define  $A^0, B^0$  and  $I^0$  by:

$$A = A^s \cup A^0, B = B^s \cup B^0, I = I^s \cup I^0.$$

We continue this decomposition for  $X = A, B$  or  $I$  as follows:

$$X^0 = X^{\partial s} \cup X^1 \text{ with } X^{\partial s} \equiv X^0 \cap \partial s.$$

If we remember that the closure of a set  $Y$  is defined by  $\bar{Y} = Y \cup \partial Y$ , we can write further:

$$X^1 = X^{\partial \bar{s}} \cup X^2 \text{ with } X^{\partial \bar{s}} \equiv X^1 \cap \partial \bar{s}$$

and

$$X^2 = X^{\partial \bar{\bar{s}}} \cup X^3 \text{ with } X^{\partial \bar{\bar{s}}} \equiv X^2 \cap \partial \bar{\bar{s}}$$

In general we define  $X^y$  for some set  $y$  by  $X^y \equiv X \cap y$ .

The micro transition rate for the transition  $(A; B) \rightarrow (A'; B')$  is denoted by  $Q_{(A;B),(A';B')}$  and the macro transition rate for the transition  $[A; B; I] \rightarrow [A'; B'; I']$  by  $\bar{Q}_{[A;B;I],[A';B';I']}$ . Finally we define  $\tilde{s} = \bar{s} \cup \partial \bar{s}$ . The rest of the notation that we need will be introduced in between the equations.

### 3.4. Macro equations

Before we can derive the macro equations, we must make some assumption about the deblocking strategy. Activation, blocking and de-activation rates are always local. By this we mean that when a node gets active, blocked or idle (without deblocking of other nodes), the transition rate only depends on the state of the nodes within a radius of length 2 from the node that changes its state. For these transitions the macro transition rates are equal to the micro transition rates, if in the macro state all nodes within radius 2 from the changing node are specified. For deblocking transitions this is no longer true if a deblocking strategy is necessary to resolve conflicts. In such cases we will assume that the deblocking strategy is such that the transition rates remain local. By this we mean:

$$Q_{(A^s \cup A^0; B^s \cup B^0), (A^{s'} \cup A^{0'}; B^{s'} \cup B^{0'})} = \bar{Q}_{[A^{\tilde{s}}; B^{\tilde{s}}; I^{\tilde{s}}], [A^{s'}; B^{s'}; I^{s'}]}$$

If  $A^{s'} \neq A^s$  and/or  $B^{s'} \neq B^s$ .

We start with the micro equations. Instead of  $\{a\}$  and  $\{b\}$  we will write  $a$  and  $b$ , if the state of only one node is meant. The micro equations can then be written as:

$$\begin{aligned} p(A; B) & \left\{ \sum_{i \in I} \lambda + \sum_{a \in A} \mu \right\} \\ & = \lambda \left\{ \sum_{\substack{a \in A \\ B \subseteq \partial(A \setminus a)}} p(A \setminus a; B) + \sum_{b \in B} p(A; B \setminus b) \right\} + \mu \sum_{a \in \bar{A}^c} p(A \cup a; B) \\ & + \sum_{\substack{a \in \partial A \setminus B \\ B \subseteq \partial(a \cup A \setminus (A \cap \partial a))}} p(a \cup A \setminus (A \cap \partial a); B \cup (A \cap \partial a)) Q_{(a \cup A \setminus (A \cap \partial a); B \cup (A \cap \partial a)), (A; B)} \end{aligned}$$



The next step in the derivation of the macro equations consists of splitting the transitions into a part which changes the  $d$ -simplex  $s$  and a part which leaves  $s$  invariant:

$$\begin{aligned}
 & p(A; B) \left\{ \sum_{i \in I^s} \lambda + \sum_{i \in I^0} \lambda + \sum_{a \in A^s} \mu + \sum_{a \in A^0} \mu \right\} \\
 &= p(A; B) \left\{ \lambda(|I^s| + |I^0|) + \mu|A^s| + \sum_{a \in A^0} \sum_{B_a \cap B^s \neq \emptyset} Q_{(A;B), (A \setminus a \cup B_a; B \setminus B_a)} \right. \\
 &\quad \left. + \sum_{a \in A^0} \sum_{B_a \cap B^s = \emptyset} Q_{(A;B), (A \setminus a \cup B_a; B \setminus B_a)} \right\} \\
 &= \sum_{\substack{a \in A^s \\ B \subseteq \partial(A \setminus a)}} \lambda p(A \setminus a; B) + \sum_{\substack{a \in A^0 \\ B \subseteq \partial(A \setminus a)}} \lambda p(A \setminus a; B) + \sum_{b \in B^s} \lambda p(A; B \setminus b) \\
 &\quad + \sum_{b \in B^0} \lambda p(A; B \setminus b) + \sum_{a \in I^s \setminus \partial A} \mu p(A \cup a; B) + \sum_{a \in I^0 \setminus \partial A} \mu p(A \cup a; B) \\
 &\quad + \sum_{\substack{a \in I^s \cap \partial A \\ B \subseteq \partial(a \cup A \setminus (A \cap \partial a))}} p(a \cup A \setminus (A \cap \partial a); B \cup (A \cap \partial a)) Q_{(a \cup A \setminus (A \cap \partial a); B \cup (A \cap \partial a)), (A; B)} \\
 &\quad + \sum_{\substack{a \in I^0 \cap \partial A^s \\ B \subseteq \partial(a \cup A \setminus (A \cap \partial a))}} p(a \cup A \setminus (A \cap \partial a); B \cup (A \cap \partial a)) Q_{(a \cup A \setminus (A \cap \partial a); B \cup (A \cap \partial a)), (A; B)} \\
 &\quad + \sum_{\substack{a \in I^0 \cap \partial A \setminus \partial A^s \\ B \subseteq \partial(a \cup A \setminus (A \cap \partial a))}} p(a \cup A \setminus (A \cap \partial a); B \cup (A \cap \partial a)) Q_{(a \cup A \setminus (A \cap \partial a); B \cup (A \cap \partial a)), (A; B)}
 \end{aligned}$$

Next, we are going to sum left- and right-hand side of this equation over  $A^0$  and  $B^0$ , i.e. over all possible states of nodes outside the  $d$ -simplex  $s$  that are compatible with  $A^s$  and  $B^s$ . We then claim that the sum over the transitions that leave  $s$  invariant gives an identity on left- and right-hand side. How can we see this? Consider a transition  $(A; B) \rightarrow (A'; B')$  which leaves  $s$  invariant:

$$\begin{aligned}
 & A \cap s = A' \cap s = A^s, \quad B \cap s = B' \cap s = B^s \\
 & \text{and } \Lambda \setminus (A \cup B) \cap s = \Lambda \setminus (A' \cup B') \cap s = I^s.
 \end{aligned}$$

The term  $p(A; B) Q_{(A;B), (A';B')}$  appears on the left-hand side of the equation for  $p(A; B)$ , but also on the right-hand side of the equation for  $p(A'; B')$ . This means that summing all equations for micro states that are compatible with  $A^s$  and  $B^s$  gives an identity for the transitions outside  $s$ . By cancelling these transitions on right- and left-hand side, we are thus left with a summation over all transitions that change the state of  $s$ :

$$\begin{aligned}
 & \sum_{A^0, B^0} p(A; B) \left\{ \lambda|I^s| + \mu|A^s| + \sum_{a \in A^0} \sum_{B_a \cap B^s \neq \emptyset} Q_{(A;B), (A \setminus a \cup B_a; B \setminus B_a)} \right\} \\
 &= \sum_{A^0, B^0} \left\{ \sum_{\substack{a \in A^s \\ B \subseteq \partial(A \setminus a)}} \lambda p(A \setminus a; B) + \sum_{b \in B^s} \lambda p(A; B \setminus b) + \sum_{a \in I^s \setminus \partial A} \mu p(A \cup a; B) \right.
 \end{aligned}$$

$$\begin{aligned}
& + \sum_{\substack{a \in I^s \cap \partial A \\ B \subseteq \partial(a \cup A \setminus (A \cap \partial a))}} p(a \cup A \setminus (A \cap \partial a); B \cup (A \cap \partial a)) Q_{(a \cup A \setminus (A \cap \partial a); B \cup (A \cap \partial a)), (A; B)} \\
& + \sum_{\substack{a \in I^0 \cap \partial A^s \\ B \subseteq \partial(a \cup A \setminus (A \cap \partial a))}} p(a \cup A \setminus (A \cap \partial a); B \cup (A \cap \partial a)) Q_{(a \cup A \setminus (A \cap \partial a); B \cup (A \cap \partial a)), (A; B)} \Bigg\}. \quad (1)
\end{aligned}$$

Next, we perform the remaining summation over  $A^0$  and  $B^0$ . In general we do this by splitting the summation into a summation over the first boundaries of  $s$  and a summation over the rest and then by interchanging the order of the summations. The result of this are the macro equations in terms of macro probabilities  $\pi$  and macro transition rates  $\bar{Q}$ . For the left-hand side we will do this explicitly, and for the right-hand side we will give the result. We will explain the meaning of all macro transition rates, however, and finally left- and right-hand side will be combined to present the complete macro equations. The first two terms on the left-hand side of Eq. (1) can immediately be rewritten, but the third term requires much more care. We can rewrite the left-hand side of Eq. (1) as:

$$\begin{aligned}
& \pi[A^s; B^s; I^s] \{ \lambda | I^s | + \mu | A^s | \} \\
& + \sum_{A^s \setminus s, B^s \setminus s} \sum_{A^3, B^3} p(A; B) \sum_{a \in A^{0s}} \sum_{B_a \cap B^s \neq \emptyset} Q_{(A; B), (A \setminus a \cup B_a; B \setminus B_a)} \\
& = \pi[A^s; B^s; I^s] \{ \lambda | I^s | + \mu | A^s | \} \\
& + \sum_{A^s \setminus s, B^s \setminus s} \pi[A^s; B^s; I^s] \sum_{a \in A^{0s}} \sum_{B_a \cap B^s \neq \emptyset} \bar{Q}_{[A^s; B^s; I^s], [A^s \setminus a \cup B_a; B^s \setminus B_a; I^s \cup a]} \\
& = \pi[A^s; B^s; I^s] \left\{ \lambda | I^s | + \mu | A^s | + \sum_{B_{0s}^s} \sum_{A^s \setminus s, B^s \setminus s} \frac{\pi[A^s; B^s; I^s]}{\pi[A^s; B^s; I^s]} \right. \\
& \quad \left. \times \bar{Q}_{[A^s; B^s; I^s], [A^s \setminus a \cup B_a; B^s \setminus B_a; I^s \cup a]} \right\} \\
& = \pi[A^s; B^s; I^s] \left\{ \lambda | I^s | + \mu | A^s | + \sum_{B_{0s}^s} \bar{Q}_{[A^s; B^s; I^s], [A^s \cup B_{0s}^s; B^s \setminus B_{0s}^s; I^s]} \right\},
\end{aligned}$$

where  $B_{0s}^s$  denotes one of the possible sets of blocked nodes in  $s$  that can be deblocked by a node  $a$  in  $\partial s$ , and  $B_a$  the set of blocked nodes in  $B$  that is deblocked by  $a$ . The rate  $\bar{Q}$  denotes the *spontaneous deblocking rate* and is given by:

$$\bar{Q}_{[A^s; B^s; I^s], [A^s \cup B_{0s}^s; B^s \setminus B_{0s}^s; I^s]} \equiv \sum_{A^s \setminus s, B^s \setminus s} \frac{\pi[A^s; B^s; I^s]}{\pi[A^s; B^s; I^s]} \bar{Q}_{[A^s; B^s; I^s], [A^s \setminus a \cup B_a; B^s \setminus B_a; I^s \cup a]}. \quad (2)$$

In a similar fashion we can rewrite the right-hand side of Eq. (1). Remember that the complement of a set  $X$  was defined as the set of nodes outside  $X$ , so  $s^c \equiv \Lambda \setminus s$ . Interdependencies between summation domains are all implicit in this notation (in an attempt to keep it

manageable), so that also any change in these interdependencies as a result of changing the order of summations must remain implicit in this notation. Apart from this, there is the handicap of the enormous size of these expressions. We will therefore not rewrite the right-hand side of Eq. (1) explicitly, but only give the result:

$$\begin{aligned}
& \sum_{a \in A^s \wedge P_+} \pi[A^s \setminus a; B^s; I^s \cup a] \bar{Q}_{[A^s \setminus a; B^s; I^s \cup a], [A^s; B^s; I^s]} \\
& + \sum_{b \in B^s} \pi[A^s; B^s \setminus b; I^s \cup b] \bar{Q}_{[A^s; B^s \setminus b; I^s \cup b], [A^s; B^s; I^s]} + \sum_{a \in I^s \setminus \partial A^s} \mu \pi[a \cup A^s; B^s; I^s \setminus a] \\
& + \sum_{a \in I^s \cap \partial A^s \wedge P_-} \pi[a \cup A^s \setminus (A^s \cap \partial a); B^s \cup (A^s \cap \partial a); I^s \setminus a] \\
& \times \bar{Q}_{[a \cup A^s \setminus (A^s \cap \partial a); B^s \cup (A^s \cap \partial a); I^s \setminus a], [A^s; B^s; I^s]} \\
& + \sum_{A_{\partial s}^s} \pi[A^s \setminus A_{\partial s}^s; B^s \cup A_{\partial s}^s; I^s] \bar{Q}_{[A^s \setminus A_{\partial s}^s; B^s \cup A_{\partial s}^s; I^s], [A^s; B^s; I^s]},
\end{aligned}$$

where  $P_+$  and  $P_-$  are defined by:

$$\begin{aligned}
P_+ : B^s \setminus \partial s^c & \subseteq \partial(A^s \setminus a) \wedge (\exists V : V \subseteq \partial s : (\forall x, y : x, y \in V : x \notin \partial y) \\
& \wedge V \cap \partial(A^s \setminus a) = \emptyset \wedge B^s \cap \partial s^c \subseteq \partial V \cup \partial(A^s \setminus a)) \\
P_- : B^s \setminus \partial s^c & \subseteq \partial(a \cup A^s \setminus (A^s \cap \partial a)) \wedge (\exists V : V \subseteq \partial s : (\forall x, y : x, y \in V : x \notin \partial y) \\
& \wedge V \cap \partial(a \cup A^s \setminus (A^s \cap \partial a)) = \emptyset \\
& \wedge B^s \cap \partial s^c \subseteq \partial V \cup \partial(a \cup A^s \setminus (A^s \cap \partial a))).
\end{aligned}$$

In words,  $P_+$  and  $P_-$  express that  $[A^s \setminus a; B^s; I^s \cup a]$  and  $[a \cup A^s \setminus (A^s \cap \partial a); B^s \cup (A^s \cap \partial a); I^s \setminus a]$  are valid macro states, given that  $[A^s; B^s; I^s]$  is.

The rate in the first term is the activation rate, and is given by:

$$\bar{Q}_{[A^s \setminus a; B^s; I^s \cup a], [A^s; B^s; I^s]} = \sum_{A_{\partial s}^s, B_{\partial s}^s} \lambda \frac{\pi[A^s \setminus a; B^s; I^s \cup a]}{\pi[A^s \setminus a; B^s; I^s \cup a]}.$$

For  $a \in A^s \setminus \partial s^c$  we have  $\bar{Q} \dots = \lambda$ .

The rate in the second term is the blocking rate and is given by:

$$\bar{Q}_{[A^s; B^s \setminus b; I^s \cup b], [A^s; B^s; I^s]} = \sum_{A_{\partial s}^s, B_{\partial s}^s} \lambda \frac{\pi[A^s; B^s \setminus b; I^s \cup b]}{\pi[A^s; B^s \setminus b; I^s \cup b]}.$$

For  $b \in \partial A^s \cap B^s$  we have  $\bar{Q} \dots = \lambda$ .

The rate in the third term is simply  $\mu$  and denotes the de-activation rate.

The rate in the fourth term denotes the *stimulated deblocking rate* and is given by:

$$\begin{aligned}
& \bar{Q}_{[a \cup A^s \setminus (A^s \cap \partial a); B^s \cup (A^s \cap \partial a); I^s \setminus a], [A^s; B^s; I^s]} \\
& = \sum_{A_{\partial s}^s, B_{\partial s}^s} \frac{\pi[a \cup A^s \setminus (A^s \cap \partial a); B^s \cup (A^s \cap \partial a); I^s \setminus a]}{\pi[a \cup A^s \setminus (A^s \cap \partial a); B^s \cup (A^s \cap \partial a); I^s \setminus a]} \\
& \times \bar{Q}_{[a \cup A^s \setminus (A^s \cap \partial a); B^s \cup (A^s \cap \partial a); I^s \setminus a], [A^s; B^s; I^s]}
\end{aligned}$$

The rate in the fifth term denotes the *spontaneous deblocking rate* and is given by:

$$\begin{aligned} \bar{Q}_{[A^s \setminus A_{\partial s}^s; B^s \cup A_{\partial s}^s; I^s], [A^s; B^s; I^s]} &= \sum_{A^s \setminus s, B^s \setminus s} \frac{\pi[a \cup A^s \setminus (A^s \cap \partial a); B^s \cup (A^s \cap \partial a); I^s \setminus a]}{\pi[A^s \setminus A_{\partial s}^s; B^s \cup A_{\partial s}^s; I^s]} \\ &\quad \times \bar{Q}_{[a \cup A^s \setminus (A^s \cap \partial a); B^s \cup (A^s \cap \partial a); I^s \setminus a], [A^s; B^s; I^s]} \end{aligned}$$

which may be rewritten in the same form as Eq. (2) if one notes that  $B_{\partial s}^s$  has been replaced by  $A_{\partial s}^s$ , since the spontaneously deblocked nodes now belong to  $A$  instead of  $B$ .

Combining the left- and right-hand side, we can write the full macro equations as:

$$\begin{aligned} \pi[A^s; B^s; I^s] &\left\{ \lambda |I^s| + \mu |A^s| + \sum_{B_{\partial s}^s} \bar{Q}_{[A^s; B^s; I^s], [A^s \cup B_{\partial s}^s; B^s \setminus B_{\partial s}^s; I^s]} \right\} \\ &= \sum_{a \in A^s \wedge P_+} \pi[A^s \setminus a; B^s; I^s \cup a] \bar{Q}_{[A^s \setminus a; B^s; I^s \cup a], [A^s; B^s; I^s]} \\ &\quad + \sum_{b \in B^s} \pi[A^s; B^s \setminus b; I^s \cup b] \bar{Q}_{[A^s; B^s \setminus b; I^s \cup b], [A^s; B^s; I^s]} + \sum_{a \in I^s \setminus \partial A^s} \mu \pi[a \cup A^s; B^s; I^s \setminus a] \\ &\quad + \sum_{a \in I^s \cap \partial A^s \wedge P_-} \pi[a \cup A^s \setminus (A^s \cap \partial a); B^s \cup (A^s \cap \partial a); I^s \setminus a] \\ &\quad \times \bar{Q}_{[a \cup A^s \setminus (A^s \cap \partial a); B^s \cup (A^s \cap \partial a); I^s \setminus a], [A^s; B^s; I^s]} \\ &\quad + \sum_{A_{\partial s}^s} \pi[A^s \setminus A_{\partial s}^s; B^s \cup A_{\partial s}^s; I^s] \bar{Q}_{[A^s \setminus A_{\partial s}^s; B^s \cup A_{\partial s}^s; I^s], [A^s; B^s; I^s]}. \end{aligned} \quad (3)$$

If the diameter  $d$  of the  $d$ -simplex  $s$  is small compared to the diameter of the whole interference graph, the number of macro equations to be solved is very much smaller than the number of micro equations. In case  $d = 0$  ( $s$  contains only one node) Eq. (3) reduces to (after dividing both sides by  $\mu$ ):

$$\begin{aligned} \pi[\emptyset; \emptyset; \{i\}] &\{\bar{Q}_{IA} + \bar{Q}_{IB}\} = \pi[\{i\}; \emptyset; \emptyset] \\ \pi[\{i\}; \emptyset; \emptyset] \bar{Q}_{AI} &= \pi[\emptyset; \emptyset; \{i\}] \bar{Q}_{IA} + \pi[\emptyset; \{i\}; \emptyset] \bar{Q}_{BA} \\ \pi[\emptyset; \{i\}; \emptyset] \bar{Q}_{BA} &= \pi[\emptyset; \emptyset; \{i\}] \bar{Q}_{IB}, \end{aligned}$$

with

$$\begin{aligned} \bar{Q}_{IA} &= \rho \sum_{b \cup e = \partial\{i\}} \pi[\emptyset; b; e \cup \{i\}] / \pi[\emptyset; \emptyset; \{i\}] \\ \bar{Q}_{IB} &= \rho \sum_{\substack{a \cup b \cup e = \partial\{i\} \\ a \neq \emptyset}} \pi[a; b; e \cup \{i\}] / \pi[\emptyset; \emptyset; \{i\}] \\ \bar{Q}_{AI} &= 1 \\ \bar{Q}_{BA} &= \sum_{a \in \partial\{i\}} \sum_{b \cup e = \partial\{i\} \setminus \{a\}} \pi[\{a\}; b \cup \{i\}; e] / \pi[\emptyset; \{i\}; \emptyset]. \end{aligned}$$

The last equation is only valid in case there are no conflicts between deblocking nodes. In general an active node can keep several neighbours blocked which cannot deblock simultaneously. In that case the deblocking rate is given by (dropping the braces around  $a$  and  $i$ ):

$$\bar{Q}_{BA} = \sum_{a \in \partial i} \sum_{b \cup e = \partial i \setminus a} \sum_{[C \cup a; D \cup b \cup i; E \cup e]} \frac{\pi[C \cup a; D \cup b \cup i; E \cup e]}{\pi[\emptyset; i; \emptyset]} \times \sum_{D_a} \bar{Q}_{[C \cup a; D \cup b \cup i; E \cup e], [C \cup D_a \cup i; D \setminus D_a \cup b; E \cup e \cup a]},$$

with  $C \cup D \cup E(\bar{i} \cup \partial i) \setminus \bar{i}$  and  $D_a \subseteq \partial a \cap D \setminus \partial C$  (one of the possible subsets of  $D$  that can be deblocked by  $a$ ).

In order to calculate  $\tau_B$  we must consider the diagonal element of  $\bar{Q}$  corresponding to the blocked state of node  $i$ , i.e.  $\bar{Q}_{BB}$ . From the blocked state  $i$  can only make a transition to the active state, so we have  $\bar{Q}_{BB} = -\bar{Q}_{BA}$ . The average lifetime of the blocked state of node  $i$  is therefore given by:

$$\tau_B = \bar{Q}_{BA}^{-1}$$

#### 4. Macro approximation

If we could solve the macro equations exactly, it would be very easy to calculate the performance measures quickly. The problem is that there are unknown macro transition rates  $\bar{Q}$  for a given  $d$ -simplex  $s$ . By making  $d$  larger one can express these unknown transition rates in terms of the macro probabilities on the larger  $d$ -simplex, but then one introduces again unknown transition rates. The  $d$ -simplex can be made larger again, but eventually one must solve all the micro equations to eliminate the unknown transition rates. In the macro approximation the unknown transition rates are approximated without expanding the  $d$ -simplex  $s$ . A term in an unknown transition rate has typically the form:

$$\frac{\pi[A^{s \cup s_c}; B^{s \cup s_c}; I^{s \cup s_c}]}{\pi[A^s; B^s; I^s]} \bar{Q}_{[A^{s \cup s_c}; B^{s \cup s_c}; I^{s \cup s_c}], [A^{s \cup s'_c}; B^{s \cup s'_c}; I^{s \cup s'_c}]}, \quad (4)$$

where we have written  $X^{s \cup s_c}$  for  $X^s \cup X^{s_c}$  and  $X^{s \cup s'_c}$  for  $X^{s'} \cup X^{s'_c}$ .  $X^s$  specifies the state of nodes inside  $s$  and  $X^{s_c}$  the state of nodes outside  $s$ :  $X^s \subseteq s$ ,  $X^{s_c} \subseteq s^c$  and similarly  $X^{s'} \subseteq s$  and  $X^{s'_c} \subseteq s'^c$  for  $X = A, B$  and  $I$ , such that  $A^s \cup B^s \cup I^s = A^{s'} \cup B^{s'} \cup I^{s'} = s$  and  $X^{s'} \neq X^s$  for  $X = A, B$  or  $I$ . In Eq. (4) the macro transition rate  $\bar{Q} \dots$  is known, since it is equal to any micro transition rate between departure and arrival micro states that are compatible with  $[A^{s \cup s_c}; B^{s \cup s_c}; I^{s \cup s_c}]$  and  $[A^{s \cup s'_c}; B^{s \cup s'_c}; I^{s \cup s'_c}]$  respectively. The macro probability  $\pi[A^s; B^s; I^s]$  is also “known” in the sense that it appears in the macro equations outside the transition rates (in fact it cancels with the corresponding term), but  $\pi[A^{s \cup s_c}; B^{s \cup s_c}; I^{s \cup s_c}]$  does not appear outside the transition rates in the macro equations and must therefore be regarded as unknown. We can interpret the quotient of the two macro probabilities as a conditional macro probability:

$$\frac{\pi[A^{s \cup s_c}; B^{s \cup s_c}; I^{s \cup s_c}]}{\pi[A^s; B^s; I^s]} = \pi[A^{s_c}; B^{s_c}; I^{s_c} | A^s; B^s; I^s],$$

which means the conditional probability of the macro state  $[A^{s_c}; B^{s_c}; I^{s_c}]$  given the macro state  $[A^s; B^s; I^s]$ . We can write this conditional probability as a product of conditional probabilities with only one node specified in the numerator if we define:

$$\begin{aligned}
 A^{s_c} &\equiv \{a_1, \dots, a_{n_A}\}, B^{s_c} \equiv \{a_{n_A+1}, \dots, a_{n_A+n_B}\} \\
 \text{and } I^{s_c} &\equiv \{a_{n_A+n_B+1}, \dots, a_{n_A+n_B+n_I}\} \\
 \Rightarrow \pi[A^{s_c}; B^{s_c}; I^{s_c} | A^s; B^s; I^s] &= \pi[a_1; \emptyset; \emptyset | A^{s \cup s_c} \setminus a_1; B^{s \cup s_c}; I^{s \cup s_c}] \\
 &\quad \times \pi[a_2; \emptyset; \emptyset | A^{s \cup s_c} \setminus \{a_1, a_2\}; B^{s \cup s_c}; I^{s \cup s_c}] \times \dots \times \pi[a_{n_A}; \emptyset; \emptyset | A^s; B^{s \cup s_c}; I^{s \cup s_c}] \\
 &\quad \times \pi[\emptyset; a_{n_A+1}; \emptyset | A^s; B^{s \cup s_c} \setminus a_{n_A+1}; I^{s \cup s_c}] \times \dots \times \pi[\emptyset; a_{n_A+n_B}; \emptyset | A^s; B^s; I^{s \cup s_c}] \\
 &\quad \times \pi[\emptyset; \emptyset; a_{n_A+n_B+1} | A^s; B^s; I^{s \cup s_c} \setminus a_{n_A+n_B+1}] \times \dots \times \pi[\emptyset; \emptyset; a_{n_A+n_B+n_I} | A^s; B^s; I^s].
 \end{aligned} \tag{5}$$

We now write the subset of nodes in a set  $X$  which are separated by a distance smaller or equal to  $d$  from node  $a_i$  as  $X_i$ . In the right-hand side of Eq. (5) appear three types of factors, i.e.  $\pi[a_i; \emptyset; \emptyset | X; Y; Z]$ ,  $\pi[\emptyset; a_j; \emptyset | U; V; W]$  and  $\pi[\emptyset; \emptyset; a_k | R; S; T]$ , that we approximate in the following way:

$$\begin{aligned}
 \pi[a_i; \emptyset; \emptyset | X; Y; Z] &\approx \pi[a_i; \emptyset; \emptyset | X_i; Y_i; Z_i] \\
 \pi[\emptyset; a_j; \emptyset | U; V; W] &\approx \pi[\emptyset; a_j; \emptyset | U_j; V_j; W_j] \\
 \pi[\emptyset; \emptyset; a_k | R; S; T] &\approx \pi[\emptyset; \emptyset; a_k | R_k; S_k; T_k].
 \end{aligned} \tag{6}$$

Now consider the term  $\pi[a_i; \emptyset; \emptyset | X_i; Y_i; Z_i]$  (for the other two terms the argument is analogous). We have two possibilities:

- 1)  $a_i \cup X_i \cup Y_i \cup Z_i \equiv C$  is a  $d$ -simplex,
- 2)  $C$  is not a  $d$ -simplex.

ad 1) In the first case there exists a mapping from the nodes in  $C$  onto nodes in  $s$ , such that the neighbour relations in  $C$  are conserved in  $s$ :

$$a_i \cup X_i \cup Y_i \cup Z_i \rightarrow a_{i'} \cup X_{i'} \cup Y_{i'} \cup Z_{i'}$$

This is because the interference graph is pointsymmetric, and because we choose  $s$  in such a way that it is a *maximal*  $d$ -simplex, i.e. if we add one node to  $s$  then it is no  $d$ -simplex anymore. If there is for a given  $d$  only one maximal  $d$ -simplex  $s$ , then  $C$  must be mapped only onto  $s$ , but if there are two maximal  $d$ -simplices  $s_1$  and  $s_2$ , it is possible that  $C$  can be mapped only onto  $s_1$  or only onto  $s_2$ . In case that  $C$  can be mapped both onto  $s_1$  and  $s_2$  we must make a choice. If there would be more than two maximal  $d$ -simplices, we could easily continue this strategy, but experience seems to show that there can never be more than two. And if there are two maximal  $d$ -simplices, there always seems to be one maximal  $(d+1)$ -simplex, two maximal  $(d+2)$ -simplices etc.

ad 2) In case  $C$  is not a  $d$ -simplex, there must be at least one pair of nodes  $a_l, a_m \in C \setminus a_i$  that is separated by a distance greater than  $d$ . Suppose  $a_l \in X_i$  and  $a_m \in Y_i$  (for the other possibilities the argument is analogous), then we can write:

$$\begin{aligned} & \pi[a_i; \emptyset; \emptyset | X_i; Y_i; Z_i] \\ &= \frac{\pi[a_l; \emptyset; \emptyset | a_i \cup X_i \setminus a_l; Y_i; Z_i]}{\pi[a_l; \emptyset; \emptyset | X_i \setminus a_l; Y_i; Z_i]} \pi[a_i; \emptyset; \emptyset | X_i \setminus a_l; Y_i; Z_i] \\ &\approx \frac{\pi[a_l; \emptyset; \emptyset | a_i \cup (X_i \setminus a_l)_l; (Y_i \setminus a_m)_l; (Z_i)_l]}{\pi[a_l; \emptyset; \emptyset | (X_i \setminus a_l)_l; (Y_i \setminus a_m)_l; (Z_i)_l]} \pi[a_i; \emptyset; \emptyset | X_i \setminus a_l; Y_i; Z_i] \end{aligned} \quad (7)$$

Now if both  $a_i \cup (X_i)_l \cup (Y_i \setminus a_m)_l \cup (Z_i)_l$  and  $a_i \cup X_i \setminus a_l \cup Y_i \cup Z_i$  are  $d$ -simplices, we are done. If they are not both  $d$ -simplices, then the approximation of Eq. (7) must be repeated for all conditional macro probabilities of which the argument is not a  $d$ -simplex. In the first set mentioned above, node  $a_m$  is no longer present and in the second set node  $a_l$  is no longer present. By repeating the approximation of Eq. (7) we can thus eliminate all pairs of nodes that are separated by distances greater than  $d$ , so that eventually  $\pi[a_i; \emptyset; \emptyset | X_i; Y_i; Z_i]$  can always be approximated by a product of conditional macro probabilities of which the arguments are  $d$ -simplices.

For the further description of the macro approximation we limit ourselves therefore now to case 1), since the same argument is valid for the factors into which the approximation of  $\pi[a_i; \emptyset; \emptyset | X_i; Y_i; Z_i]$  in case 2) can be decomposed.

We will now explain why pointsymmetry is essential for the macro approximation. Since the mapping  $a_i \cup X_i \cup Y_i \cup Z_i \rightarrow a_{i'} \cup X_{i'} \cup Y_{i'} \cup Z_{i'}$  from  $C$  onto  $s$  conserves the neighbour relations, we must have for the exact macro probabilities:

$$\pi[a_i; \emptyset; \emptyset | X_i; Y_i; Z_i] = \pi[a_{i'}; \emptyset; \emptyset | X_{i'}; Y_{i'}; Z_{i'}],$$

because of the pointsymmetry of the interference graph. This equality makes it possible to replace approximation (6) by:

$$\pi[a_i; \emptyset; \emptyset | X; Y; Z] \approx \tilde{\pi}[a_{i'}; \emptyset; \emptyset | X_{i'}; Y_{i'}; Z_{i'}], \quad (8)$$

where we still must sum over the possible states of nodes that are not specified in  $s$ :

$$\begin{aligned} & \tilde{\pi}[a_{i'}; \emptyset; \emptyset | X_{i'}; Y_{i'}; Z_{i'}] \\ &= \frac{\sum_{\substack{[a_{i'} \cup X_{i'} \cup A; Y_{i'} \cup B; Z_{i'} \cup I] \\ C' \cup A \cup B \cup I = s}} \tilde{\pi}[a_{i'} \cup X_{i'} \cup A; Y_{i'} \cup B; Z_{i'} \cup I]}{\sum_{\substack{[X_{i'} \cup A'; Y_{i'} \cup B'; Z_{i'} \cup I'] \\ C' \setminus a_{i'} \cup A' \cup B' \cup I' = s}} \tilde{\pi}[X_{i'} \cup A'; Y_{i'} \cup B'; Z_{i'} \cup I']}, \end{aligned}$$

with  $C' \equiv a_{i'} \cup X_{i'} \cup Y_{i'} \cup Z_{i'}$ .  $\tilde{\pi}[A^s; B^s; I^s]$  is the macro approximation of  $\pi[A^s; B^s; I^s]$  that we get by substituting  $\tilde{\pi}$  for  $\pi$  in the macro equations (3) and by applying approximation (8) to the unknown macro transition rates  $\bar{Q}$ . The result is a set of nonlinear equations for the approximate macro probabilities  $\tilde{\pi}$ . The obvious method of solution is iteration: choose initial values for the unknown macro transition rates  $\bar{Q}_0$  and use the initial values to find the solution

$\tilde{\pi}_0 = f(\tilde{Q}_0)$  of the macro equations. Calculate by means of  $\tilde{\pi}_0$  and Eq. (8) new values for the unknown macro transition rates  $\tilde{Q}_1 = g(\tilde{\pi}_0)$  and solve the macro equations again to find  $\tilde{\pi}_1 = f(\tilde{Q}_1)$ . Next determine  $\tilde{Q}_2 = g(\tilde{\pi}_1)$  and subsequently  $\tilde{\pi}_2 = f(\tilde{Q}_2)$ . Continue the iteration until the fixed point has been sufficiently close approximated. In the fixed point we have:

$$\tilde{\pi} = f(\tilde{Q}) \text{ and } \tilde{Q} = g(\tilde{\pi}),$$

or equivalently:

$$\tilde{\pi} = f(g(\tilde{\pi})) \text{ and } \tilde{Q} = g(f(\tilde{Q})).$$

The question is of course whether such a fixed point exists, whether it is unique and whether the iteration converges. To none of these questions a definite answer can be given at the moment. This is the usual situation for this type of approximation. The macro approximation belongs to the family of so called ‘fixed point approximations’, see [6] and [3]. Mitrani [3] makes the following remark on the question of existence and uniqueness of the fixed point: “As far as existence and uniqueness of the solution of the fixed-point equation (18) is concerned, the situation is familiar: nothing can be proved formally, but in practical cases the equation can be solved easily and yields good approximations”. And about the question of convergence he remarks: “Sometimes (fortunately quite rarely) that sequence will not converge, even though the solution exists. Then a different search method should be used”. In the next section we will see that the iteration always converges for the examples that we considered (with one exception though). There are also a few questions that are specific for the macro approximation. The right-hand side of Eq. (5) is not unique: there are many decompositions possible by choosing another labelling for the nodes in  $A^{s_c}$ ,  $B^{s_c}$  and  $I^{s_c}$ . One may wonder now whether substitution of the approximation (6) in the right-hand side of Eq. (5) is invariant for every possible decomposition. This is indeed the case. For a proof and discussion of other properties (like pointsymmetry breaking) of the macro approximation, the reader is referred to [5].

## 5. Results of the macro approximation

First we studied the macro approximation for a ring of 8 nodes, where the results could still be compared to an exact solution. In Fig. 1 we see the throughput per node as a function of  $\rho$ . The upper curve is the exact solution, the macro approximation for  $d = 3$  and  $d = 2$ , whereas the lower curve is for  $d = 1$ .

For  $d = 1$  the macro approximation is in this case unambiguous, because the spontaneous deblocking rate is known to be equal to  $\mu$  and does not need to be approximated therefore. We see that the macro approximations for  $d = 3$  and  $d = 2$  coincide with the exact solution. The maximal errors are  $10^{-3}$  for  $0 \leq \rho \leq 2$ . For  $d = 3$  we must solve 22 nonlinear equations and 12 for  $d = 2$ . For  $d = 3$  we also checked what the errors are for  $\rho > 2$ . The error increases slowly till 1% at  $\rho = 7$  and till 3% at  $\rho = 100$ . Then it drops slowly till 2% at  $\rho = 1000$ . Theoretically one can show that the error for both  $d = 3$  and  $d = 2$  goes to zero as  $\rho \rightarrow \infty$ . The macro approximation for  $d = 1$  is less accurate, but it needs only 5 nonlinear equations to be solved. In Fig. 2 we see the average reciprocal blocked lifetime  $\tau^{-1}$  as a function of  $\rho$ .



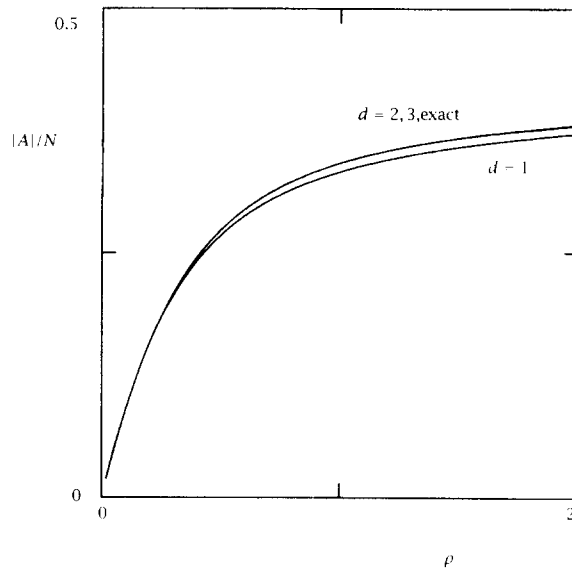


Fig. 1. Throughput per node in macro approximation as a function of  $\rho$  for a ring of 8 nodes.

The upper curve is the macro approximation for  $d = 3$  and  $d = 2$  and also the exact solution, whereas the lower curve is the macro approximation for  $d = 1$ . Actually, one can see that the upper curve consists of two curves. This is because the macro approximations for  $d = 3$  and  $d = 2$  coincide, whereas the exact solution deviates (it is the lowest one of the double curve). The slight deviation is probably an indication that the macro approximations for  $d = 3$  and  $d = 2$  approximate 1-node macro probabilities slightly better than 3-node macro probabilities.

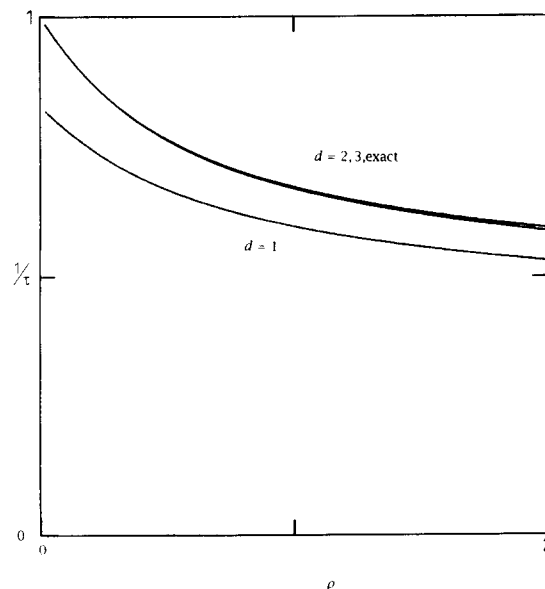


Fig. 2.  $1/\tau$  as a function of  $\rho$  in macro approximation on a ring of 8 nodes.

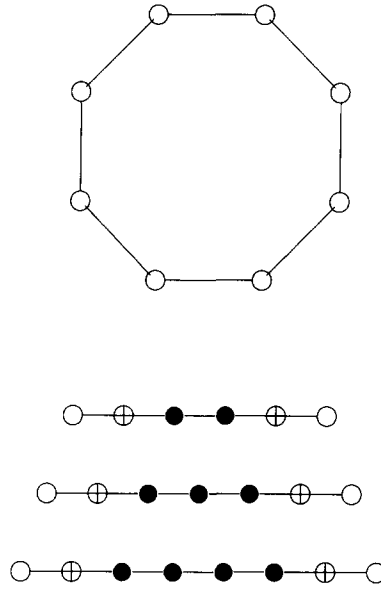


Fig. 3. Above: topology of the ring; below: maximal 1-, 2- and 3-simplices with their boundaries.

We see further that the curve for  $d = 1$  deviates strongly from the exact solution; it does not even give the correct value for  $1/\tau$  in the limit  $\rho \rightarrow 0$ . This is not a big surprise, since the 3-node macro probabilities which are necessary for the calculation of  $1/\tau$  cannot be approximated very well by the macro approximation on a maximal 1-simplex. In Fig. 3 we see an example of the ring topology and the maximal 1-, 2- and 3-simplices on the ring; the simplices are indicated with black dots, and their boundaries with a +.

Next we studied the macro approximation on a triangular dodecahedron. This is a dodecahedron where the vertices have been replaced by triangles, see Fig. 4.

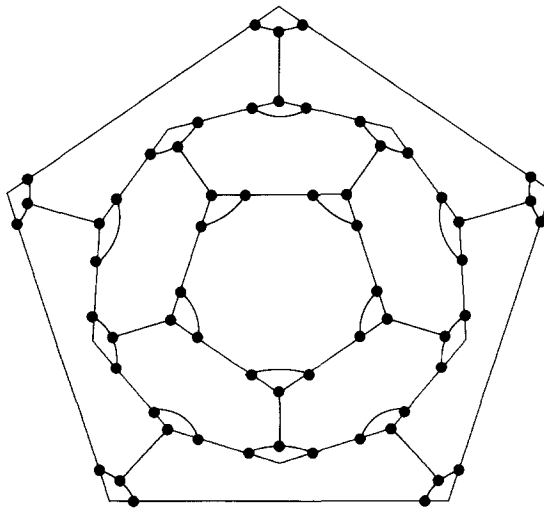


Fig. 4. Triangular dodecahedron.

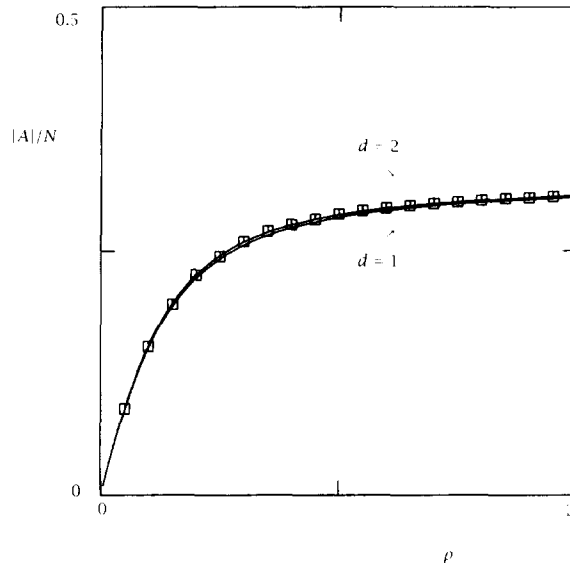


Fig. 5. Throughput per node as a function of  $\rho$  on a triangular dodecahedron.

Since the dodecahedron has 15 vertices, this interference graph has 60 nodes and more than  $2^{60}$  micro states. An exact solution of the (micro) equilibrium equations is therefore out of the question, and we have to rely on simulations to judge the results of the macro approximation. As deblocking strategy we choose: always deblock the maximal number of blocked nodes and if this can be done in more than one way, select each alternative with equal probability. We also tried another deblocking strategy which is almost the same except when we have a triangle with one active and two blocked nodes where one of the blocked nodes has an idle and the other one a blocked neighbour outside the triangle. If the active node becomes idle, the blocked node with the idle neighbour becomes active (in the other strategy each blocked node could have become active with probability  $\frac{1}{2}$ ). The idea was that such a strategy might shorten the average blocked lifetime, but we could not see any difference with the other strategy for  $0 \leq \rho \leq 2$ .

In Fig. 5 we see the throughput per node as a function of  $\rho$ . The upper curve belongs to  $d = 2$ , the lower one to  $d = 1$  and the squares denote simulation points. We see that the curve which belongs to  $d = 2$  is the best fit through the squares, but almost coincides with the lower curve. For  $d = 1$  the macro approximation has an ambiguity in this case which can be removed by renormalising the spontaneous deblocking rate, but this had no visible effect on the quality of the macro approximation (see [5]). Nevertheless, internal pointsymmetry is not broken for  $d = 1$ , which is an example of the anomaly of  $d = 1$ . For each square the computer had to solve 30 nonlinear equations for  $d = 2$ , which took about 1 s on a simple XT-pc and 12 nonlinear equations for  $d = 1$ , which took much less than 1 s on the same machine, whereas the simulation took 1 h on the same machine. The curves were calculated for much more  $\rho$ -values than just the squares, however. In Fig. 6 we see  $1/\tau$  as a function of  $\rho$ .

The squares denote the simulation points again, whereas the upper curve belongs to  $d = 2$  and the lower one to  $d = 1$ . We see again that the curve for  $d = 1$  is a very poor approximation of  $1/\tau$ , whereas the curve for  $d = 2$  is a good one. The errors in the simulation points for  $1/\tau$

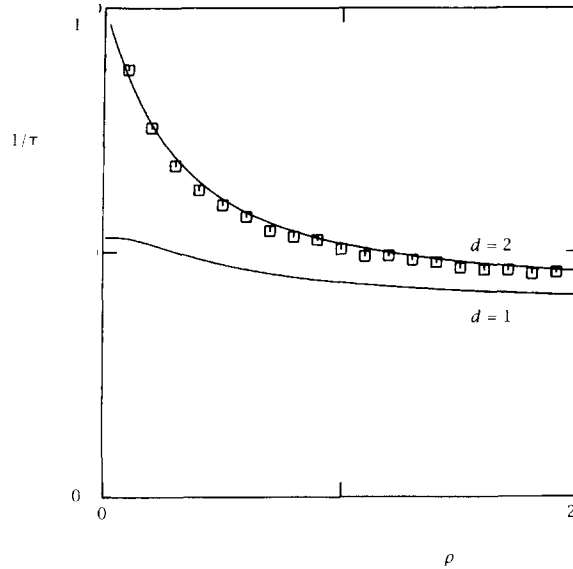


Fig. 6.  $1/\tau$  as a function of  $\rho$  on a triangular dodecahedron.

are larger than for  $|A|/N$ , because the end of the observation time of a blocked node usually does not coincide with the end of its blocked state. This problem can only occur if the observed node is in the blocked state at the end of the observation time. By counting the last blocked period of the node under observation as a full blocked lifetime, one introduces an error. On the other hand, it is always clear whether a node is in the active state or not, so that there is no similar problem in the measurement of the throughput per node. In Fig. 7 we see the two maximal 1-simplices and in Fig. 8 the maximal 2-simplex on the triangular dodecahedron. The simplices are depicted in black, and their boundaries with circles with +. In Fig. 9 we see an illustration of the macro approximation for  $d = 2$  on the triangular dodecahedron.

More precisely, it is an illustration of approximation (6) for a group of macro transition rates that appear in the macro equations for the triangular dodecahedron in case  $d = 2$ . Nodes whose states are specified are indicated by black dots, nodes over whose states a summation is carried out are indicated by double circles, and nodes whose states are unspecified by small white dots. In this case only 6 nodes need to be specified in the expression for the macro transition rates, but there are also other rates for which more nodes need to be specified.

If we denote the states of the nodes 1, 2, 3, 4 within the maximal 2-simplex by  $X_1, X_2, X_3, X_4$  and the states of the nodes 7 and 8 outside the maximal 2-simplex by  $Y_7$  and  $Y_8$  respectively, then the graphical representation of the macro approximation in Fig. 9 is equivalent to:

$$\begin{aligned} \tilde{\pi}[Y_8 Y_7 X_1 X_2 X_3 X_4] &= \frac{\tilde{\pi}[Y_8 X_1 X_2 X_3 X_4] \cdot \tilde{\pi}[Y_7 X_2 X_3 X_4]}{\sum_{Y_7'} \tilde{\pi}[Y_7' X_2 X_3 X_4]} \\ &= \frac{\tilde{\pi}[X_1 X_2 X_3 X_4] \cdot \tilde{\pi}[Y_8 X_2 X_3 X_4] \cdot \tilde{\pi}[Y_7 X_2 X_3 X_4]}{\sum_{Y_8'} \tilde{\pi}[Y_8' X_2 X_3 X_4] \cdot \sum_{Y_7'} \tilde{\pi}[Y_7' X_2 X_3 X_4]}. \end{aligned}$$

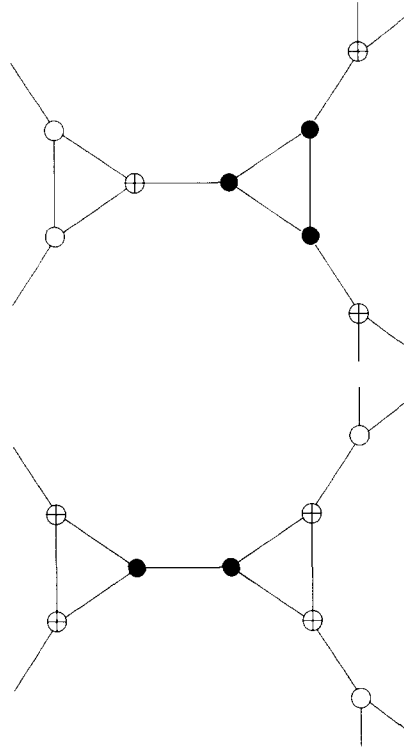


Fig. 7. The two maximal 1-simplices with their boundaries on the triangular dodecahedron.

We conclude this section with some results on another pointsymmetric graph: the torus. Figs. 10 and 11 show throughput and  $\tau^{-1}$  compared with experiments on a  $8 \times 8$  torus. Clearly the conclusions with respect to the difference between  $d = 2$  and  $d = 1$  are the same as above. Removing the ambiguity in the spontaneous deblocking rate for  $d = 1$  now has a visible effect on the quality of the approximation, however. For  $d = 1$  the spontaneous deblocking rate can be calculated as the conditional probability that a blocked node in a maximal 1-simplex has one

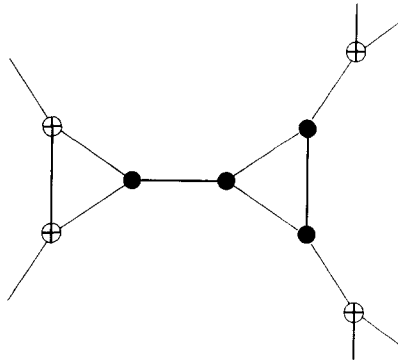


Fig. 8. The maximal 2-simplex with its boundary on the triangular dodecahedron.

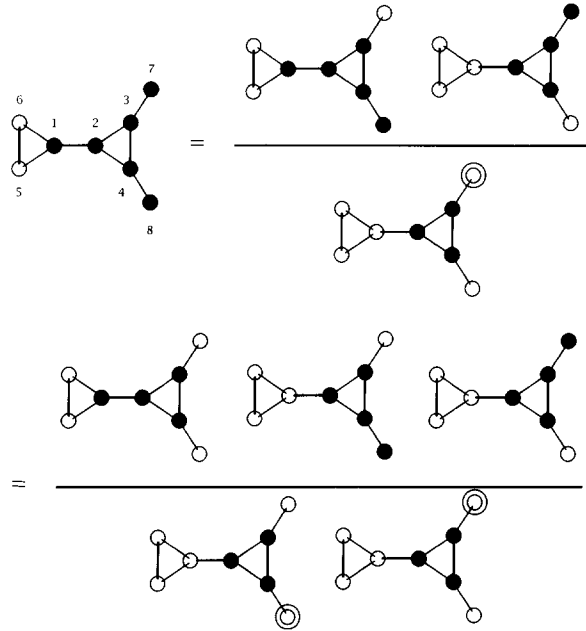


Fig. 9. Illustration of the macro approximation for  $d = 2$  on the triangular dodecahedron.

active neighbour outside the 1-simplex, but also as 1-the conditional probability that it has more than 1 active neighbour outside the 1-simplex. The exact conditional probabilities add up to 1, but the macro approximations for  $d = 1$  do not. Renormalising them in such a way that they add up to 1 again gives rise to the middle curve  $d = 1^*$  in Fig. 10 which can be clearly

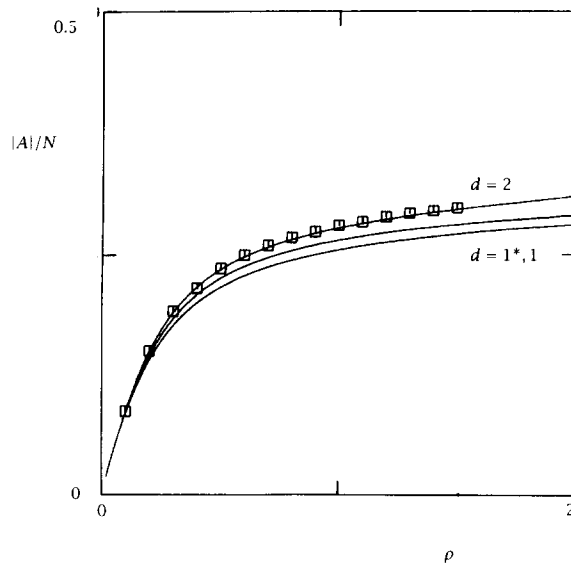


Fig. 10. Throughput per node as a function of  $\rho$  on a  $8 \times 8$  torus.

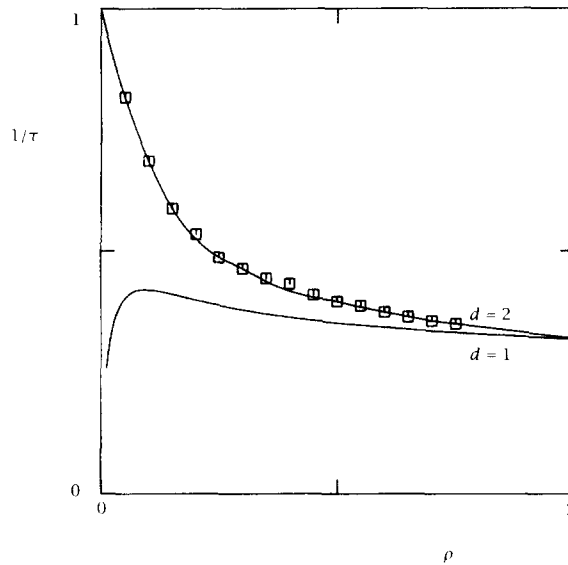


Fig. 11.  $1/\tau$  as a function of  $\rho$  in macro approximation on a  $8 \times 8$  torus.

distinguished from the lower curve corresponding to  $d = 1$ , where one of the unrenormalised conditional probabilities was used (for more details see [5]). The only case where the iteration did not converge was  $d = 1^*$ . In this case we use a standard library routine (NAG) instead, which uses the Newton–Raphson method. Note that the throughput per node is almost the same as on the triangular dodecahedron. The reciprocal blocked average lifetime  $\tau^{-1}$  is quite different however, as we can see in Fig. 11. We did not plot the curve for  $d = 1^*$  there, since it was about as bad as  $d = 1$ . We also ran the simulation on a  $11 \times 11$ -torus, but could see no significant difference with the  $8 \times 8$ -torus.

## 6. Conclusions

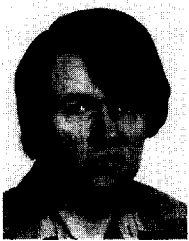
We have seen that the macro approximation is very accurate for  $d = 2$  and  $d = 3$ , and less accurate for  $d = 1$  with respect to the throughput per node; for the calculation of  $\tau$  the macro approximation is clearly inadequate for  $d = 1$ .

For the torus 58 nonlinear equations with a total length of roughly the size of this article must be solved for  $d = 2$ . It seems fair, therefore, to state that the use of the macro approximation is limited by the number of neighbours per node in the interference graph. For instance, we could not apply this approximation to a triangular lattice where each node has 6 neighbours, because of the enormous complexity. On the other hand, it seems likely that the macro approximation can also be applied to interference graphs that are not strictly pointsymmetric, like lines, stars,  $H$ -trees etc., as long as they are large enough. This is because boundary effects usually become negligible in the large-volume limit. We studied the accuracy of the macro approximation only experimentally. We did investigate whether the magnitude of the error in the throughput per node could be attributed to the error involved in approximation (6).

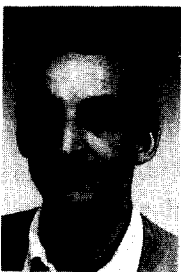
The latter error turned out to be 10 times higher than the former, so apparently many errors must cancel in the macro approximation. We are not yet able to explain this phenomenon theoretically.

## References

- [1] N. Biggs, *Algebraic Graph Theory* (Cambridge University Press, Cambridge, 1974).
- [2] A. Brandwajn and Y.L. Jow, Tandem exponential queues with finite buffers, in: T. Hasegawa, H. Takagi and Y. Takahashi, eds., *Computer Networking and Performance Evaluation* (Elsevier, Amsterdam, 1986) 245–257.
- [3] I. Mitrani, Fixed-point approximations for distributed systems, in: G. Iazeolla, P.J. Courtois and A. Hordijk, eds., *Mathematical Computer Performance and Reliability* (Elsevier, Amsterdam, 1984) pp. 245–257.
- [4] E. Pinsky and Y. Yemini, A statistical mechanics of some interconnection networks, in: E. Gelenbe, ed., *Performance 84* (Elsevier, Amsterdam, 1984) 147–158.
- [5] J.P. Veltkamp, Cost Functions for Interference Systems and VLSI High-Level Synthesis. Ph.D. thesis, University of Twente, Netherlands, 1988.
- [6] R.J. Walstra, Fixed point approximations in models of networks of queues, in: T. Hasegawa, H. Takagi and Y. Takahashi, eds., *Computer Networking and Performance Evaluation* (Elsevier, Amsterdam, 1986) 149–159.
- [7] Y. Yemini, A statistical mechanics of distributed resource sharing mechanisms, Technical report, Department of Computer Science, Columbia University, 1982.



**J. Peter Veltkamp** received an M.Sc. degree in physics from the University of Utrecht, The Netherlands, in 1983 and a Ph.D. degree in computing science from the University of Twente in 1988. Since 1989 he is an assistant professor in the Department of Mathematics and Computing Science at Eindhoven University of Technology. His current research interests include the synthesis of scalable parallel algorithms and the performance analysis and optimisation of their implementations as asynchronous digital VLSI circuits.



**Ruud van Damme** is from origin a theoretical physicist: his M.Sc. degree as well as his Ph.D. thesis (1984) were written on the subject of particle physics at the University of Utrecht. After this period he gradually changed his fields of interests towards mathematics: theory of dynamical systems (universities of Leiden and Twente) and queueing theory at the Department of Computer Science in Twente (of which this paper is a result). At this moment he is an assistant professor in the numerical analysis group at the Department of Applied Analysis at the University of Twente.